Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (cancelled)

2. (currently amended) The method as claimed in claim ~~1~~16, wherein said ~~client~~ sending a request from the client computer to the shared storage medium ~~said data version to said shared medium~~ comprises sending ~~the~~ a null-valued ~~data~~ version number ~~in the case in which~~ when said selected data element is not stored ~~in~~ on said client ~~memory~~ computer, and wherein said sending of one of a first reply and a second reply comprises sending ~~and said shared medium replying~~ to said client computer ~~with~~ a copy of ~~said~~ the selected data element and a respective ~~data~~ version number for storage on said client computer.

3. (currently amended) The method as claimed in claim ~~1~~16, wherein said request ~~for said data element~~ contains an address range defining said selected data element on said shared medium.

4. (currently amended) The method as claimed in claim 3, wherein said address range comprises non-contiguous storage blocks.

5. (currently amended) The method as claimed in claim ~~1~~16, wherein said client computer communicates with said shared medium through a network block device driver.

6. (currently amended) The method as claimed in claim ~~1~~16, wherein said shared medium is a server memory storage space.

7. (cancelled)

8. (currently amended) The method as claimed in claim ~~7~~18, wherein the request comprises information identifying multiple selected data elements ~~if said data elements being modified are~~ associated with multiple separate data structures each containing one of multiple respective version numbers~~data version information~~, and further comprising~~,~~ creating a new single data structure in said master list associated with said selected data elements ~~modified~~ and

removing said multiple separate data structures from said master list after said selected data elements is accessed by said client computer.

9. (currently amended) The method as claimed in claim 18~~7~~, further comprising updating the master list with a new version number after the selected data element is modified~~wherein said initial version state is an initial version number and wherein said initial version number is incremented to obtain said new version state~~.

10. (currently amended) The method as claimed in claim ~~7~~18, wherein said master list comprises a list of data structures which is a double linked binary tree list.

11. to 15. (cancelled)

16. (new) A method for executing a common task in a clustered computing environment comprising a plurality of computers interconnected to collaborate on said common task, said plurality of computers including at least a client computer and a shared storage medium storing common data, the method comprising:

maintaining, at the shared storage medium, a main list including information identifying data elements stored in the shared storage medium and a respective version number of each data element;

maintaining, at each client computer, a local list containing, for each one of the data elements previously accessed from the shared storage medium, information identifying a corresponding original data element in the shared storage medium and a respective current version number of the data element stored in the client computer;

when an application executing on a given client computer attempts to access a selected data element, sending a request from the client computer to the shared storage medium, the request including the information identifying the selected data element and the respective current version number;

upon receipt of the request by the shared storage medium, matching the current version number received from the client computer with the respective version number in the main list;

based on the matching, sending to the client computer one of: a first reply including a confirmation that the current data version of the selected data element is valid when the current version number received from the client computer matches the version number associated with the data element in the main list; and a second reply including a new copy of the selected data element and a respective new version number when the version number received from the client computer does not match the version number in the main list; and

upon receipt of the first reply from the shared storage medium, executing the common task at the client computer using the data element stored in the client computer; and

upon receipt of the second reply, modifying the data element stored in the client computer using the received data element prior to executing the common task.

17. (new) In a clustered computing environment comprising a plurality of computers connected to access common data stored on a shared storage medium, a method of managing access to the common data by each computer, the method comprising:

providing a respective client for each computer, the client being operative to:

maintain a local list including, for each data element previously accessed from the shared storage medium and stored in a local cache of the client, information identifying a corresponding original data element in the shared storage medium and a respective current version number of the data element stored in a local cache of the client; and

when an application executing on the computer attempts to access a selected data element:

send a request to the shared storage medium, the request including the information identifying the selected data element and the respective current version number associated with the selected data element; and

receive from the shared storage medium one of: a first reply including a confirmation that the current version number of the selected data element is valid; and a second reply including an updated content of the selected data element along with a respective new version number.

18. (New) In a clustered computing environment comprising a plurality of computers connected to access common data stored on a shared storage medium, a method of managing access to the common data by each computer, the method comprising:

maintaining a master list including, for each data element of the common data previously accessed by any one of the computers, respective information identifying the original data element in the shared storage medium and a respective current version number of the data element; and

receiving a request from a respective client executing on a given computer and attempting to access a selected data element, the request including information identifying ~~a~~ the selected data element and a respective version number of the selected data element;

determining whether the version number of the selected data element contained in the request matches the current version number of the data element;

when the version number contained in the request matches the current version number, sending a first reply to the client, the first reply including a confirmation that the version number of the selected data element contained in the request is valid; and

when the version number contained in the request does not match the current version number, sending a second reply to the client, the second reply including the selected data element along with the respective current version number.

19. (new) The method as claimed in claim 17, wherein the client computer is operative to receive from the shared storage medium a copy of the selected data element and a respective version number when said selected data element is not stored on said client computer, and upon a null-valued version number being sent in the request.

20. (new) The method as claimed in claim 17, wherein said request contains an address range defining said selected data element on said shared medium.

21. (new) The method as claimed in claim 20, wherein said address range comprises non-contiguous storage blocks.

22. (new) The method as claimed in claim 17, wherein said client computer communicates with said shared medium through a network block device driver.